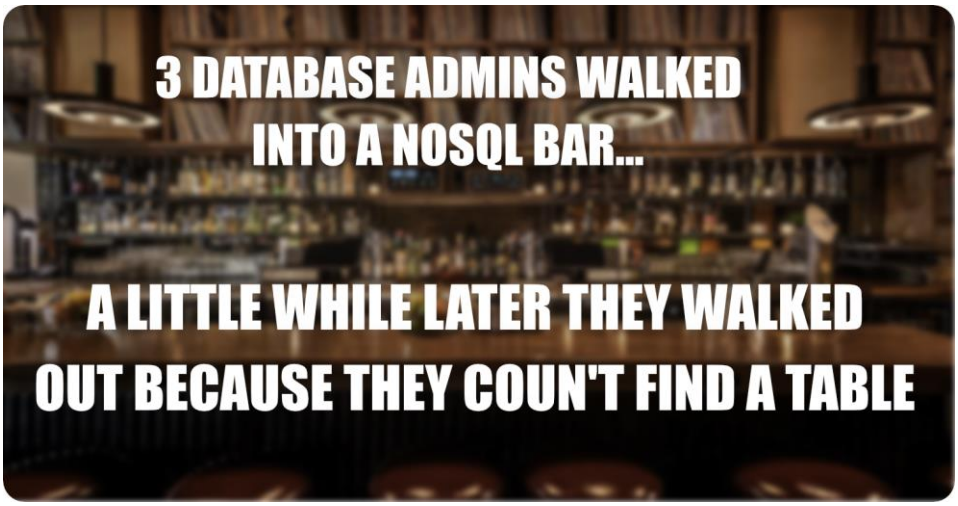

ASSIGNMENT 1

ITRI 623

ENRICO DREYER
31210783

Table of Contents

Table Of Figures

Introduction

The traditional relational database management system is a way of storing structured data in business and web applications (Strauch et al., 2011). With the new thinking of "one size fits all" databases, has led to the creation of a variety of alternative databases. The new datastores commonly fall under the new term NoSQL. This assignment will give a brief overview on what NoSQL is and the four categories of NoSQL databases including: column family stores, key-value pairs, document store, and graph databases.

What is NoSql?

According to Tiwari (2011) the word NoSql is the combination of the words No and SQL. It implies that NoSQL is the counter of SQL technology. NoSQL has become the term for any data stores and database that does not follow the well-established and popular RDBMS principles (Tiwari, 2011). NoSql is not a single technology or single product but represents a collection of diverse concepts of data manipulation and storage.

The definition of NoSql according to w3schools (2021) is an approach to the design of a database, with the goal of a vast diversity of data such as documents, multimedia, and external files to name a few. The purpose of NoSQL is to develop a wat of handling specific data models and to have flexible schemas to develop modern applications.

NoSQL does not follow the Relational Database Management System rules, thus also being called a non-relational database. NoSQL is known for its ease of development, high performance and scaling in terms of performance. The most common examples are HyperGraphDB, MongoDB and Neo4J.

Following will be a summary of the four categories of NoSQL.

Column Family Stores

According to Curé et al. (2013), column family stores builds on key-value pair types, where each key-value pair is like a row, where a column family is almost the same as a table in a relational database model. Examples include Apache Cassandra and Apache HBase (fullstackpython, 2021).

Column Family Stores allows the storage of data with a key that is mapped to a value, where the values are stored in a family of columns (Curé et al., 2013). The key to the row must be unique in the column family, but the same key can be used again in another column family.

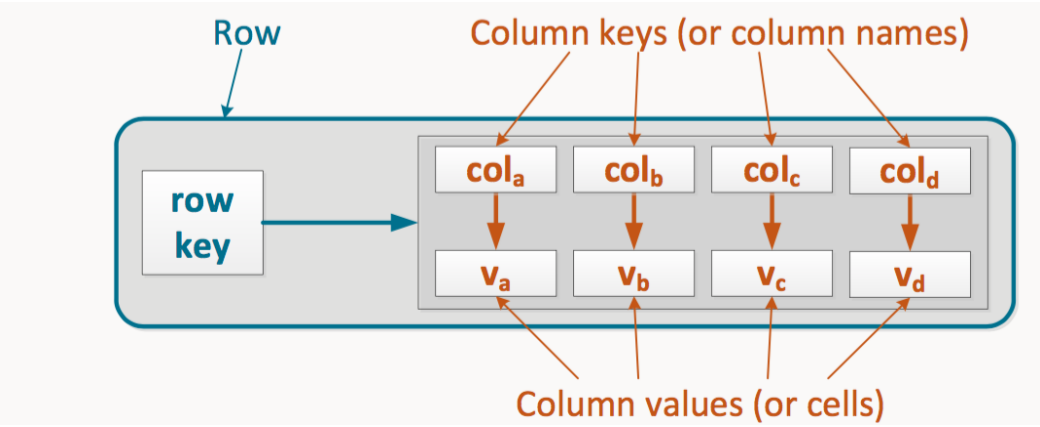

Figure 1: Column Family (studio3t, 2021)

Key-value pairs works with two columns, the hashed key and the value (Mason, 2015). The values in the column can be complex data types or simple text. Data is then retrieved by matching the key to the data that corresponds to it (studio3t, 2021).

One of the biggest advantages of using this type of NoSQL database is that a new type of data can easily be added as a new key value pair (Mason, 2015). Examples include Voldemort, Dynamo and Cache.

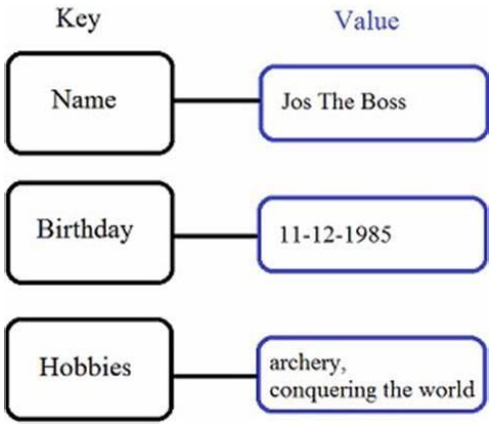

Figure 2: Key-Value Stores (Meysman, 2016)

Document Stores

As the name suggests, document-orientated databases stores information in the form of a document (Strauch et al., 2011). In a relational database, when the database is in third normal form, data is normalized to reduce redundancy of data, but in a document-orientated NoSQL database the data is stored hierarchically, semi-structured and denormalized (Mason, 2015).

Not only can additional information be added to a particular document, but you can reference another collection of documents (Mason, 2015), this is similar to a foreign key used in a RDBMS. A popular example is MongoDB.

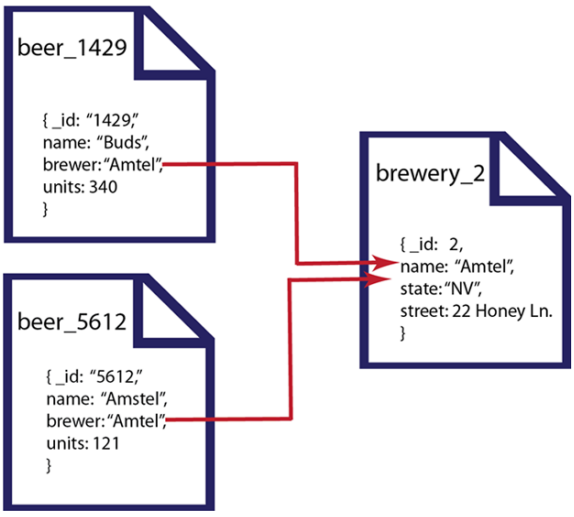

Figure 3: Document-oriented database (Braga, 2018)

Graph Databases

Graph Databases is the way of sorting relations between entities in the most efficient way (Meysman, 2016). This is most popularly used in social networks, graph databases or capital asset clusters, the reason being that the data is highly interconnected (studio3t, 2021).

The main components in a graph database are:

Nodes: This is the entities themselves, for example the people in the social network.

Edges: This is the relationships between the nodes and is represented by a line. An edge can have a direction, represented by an arrow.

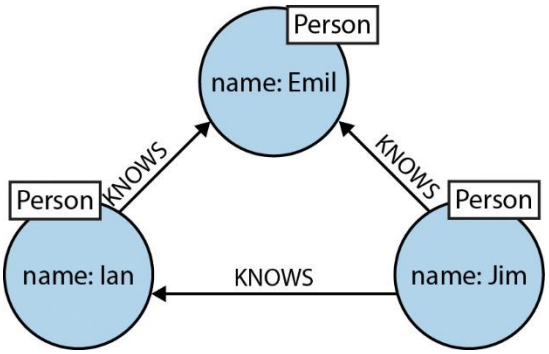

Figure 4: Graph Database (Sasaki, 2018)

Conclusion

In this assignment NoSQL databases was explained, as well as the four categories of NoSQL. NoSQL databases play an important role in Big Data in terms of storing data and getting large amounts of data. NoSQL can scale horizontally and vertically, thus making them more flexible that only vertically scaling in a RDBMS (Mason, 2015). Even though can provide an edge on performance, some are cautions in terms of data consistency (studio3t, 2021). As shown in this assignment, different scenarios require different types of NoSQL databases to emphasise on each strong point.

Referencing

Braga, B. (2018). *Copy/Export a Cloud Firestore Database*. https://blog.cloudboost.io/copy-export-a-cloud-firestore-database-388cde99259b

Curé, O., Lamolle, M., & Duc, C. L. (2013). Ontology based data integration over document and column family oriented NOSQL. *arXiv preprint arXiv:1307.2603*.

fullstackpython. (2021). *NoSQL Data Stores*. https://www.fullstackpython.com/no-sql-datastore.html

Mason, R. T. (2015). NoSQL databases and data modeling techniques for a document-oriented NoSQL database. Proceedings of Informing Science & IT Education Conference (InSITE),

Meysman, A. (2016). *NoSQL Database Types*. https://dzone.com/articles/nosql-database-types-1

Sasaki, B. M. (2018). *Graph Databases for Beginners: Why a Database Query Language Matters (More Than You Think)*. https://neo4j.com/blog/why-database-query-language-matters/

Strauch, C., Sites, U.-L. S., & Kriha, W. (2011). NoSQL databases. *Lecture Notes, Stuttgart Media University*, *20*, 24.

studio3t. (2021). *The Main NoSQL Database Types*. https://studio3t.com/knowledge-base/articles/nosql-database-types/

Tiwari, S. (2011). *Professional nosql*. John Wiley & Sons.

w3schools. (2021). *What is NoSQL*. https://www.w3schools.in/definition/what-is-nosql/